POSTER 23#

Cyber Java Monopoly: Game-based approach of collaborative programming language learning

I-Han Hsiao, Yi-Ling Lin
University of Pittsburgh
School of Information Sciences
Pittsburgh, PA 15260
{ihh4@pitt.edu, yil54@pitt.edu}

1. Introduction & Motivation

Game-based learning is one of the modern pedagogical approaches beginning to gain attention in education. The intention of adopting games in learning is to offer an additional option for online teaching and to provide students with possibilities to acquire skills and competencies. The teacher's perspective game-based learning motivates students to become active learners while offering alternative ways to interact and communicate with the system. On the other hand, it's fun to learn by doing from students' points of view [1]. UniGame Project [2] tests diverse educational games in various subjects and promotes digital game-based learning in Europe. The field of Medicine [3][4] highlighted the usefulness of specifying problem scenarios which are placed within a game framework [5]. However, in traditional game-based learning teachers have to locate resources for the games and the creation of game stories and curriculum-relevant game content is an enormous workload for teachers and learning content providers [6]. Therefore, the question that we tempt to answer is: Do students learn more through highly engaged annotating activity? Is game-based learning a good and efficient instructional strategy for supporting programming language learning? Is it proper to apply game-based approach into collaborative programming language learning?

Previous research [7] has shown that the student community is able to distinguish good and bad annotations and valuable ratings and comments. Based on peer review, students improve the annotation quality during a re-annotation process. The phase by phase activity helps students understand the subject and contributes to their learning in a programming language course. Thus, this research intends to blend peer review and annotation mechanisms into the programming language courses curriculum. In addition, in order to motivate students to interact with each other and the system itself, we introduce gaming elements to stimulate interaction within student community.

Furthermore, the structural phases are essentially clear and predefined to comply with learning objectives. Therefore, the novelty of this research is to combine game rules in existing structure. It not only corresponds to one of the key characteristics of game-based learning, it also removes the burden from the teachers to locate the resources to the games and create the contents; at the same time, it allows teachers to focus on other pedagogical tasks.

2. Game Design & Rules

We extract the essence from the well-known monopoly board game and construct a Cyber Monopoly map. Each cell represents a domain concept and it consists of a set of un-annotated examples (Figure 1). In Monopoly game, one of the main characteristics is the turn taking mechanism; each player rolls dice and moves resulting in economic activities. In our Cyber Java Monopoly game, no dice rolling is required. Each player (team) should move accordingly through the sequence of domain topics, which also corresponds to the course syllabus schedule. Players (P) are teams of students. Students are randomly and anonymously divided into two groups (team A and team B, Equation 1). The Power of Editing the annotations and re-annotations will be equally shared among students within the team. Each team strives to dominate the cells by achieving higher acquisition scores. There are three phases to acquire acquisition scores (Figure 2).

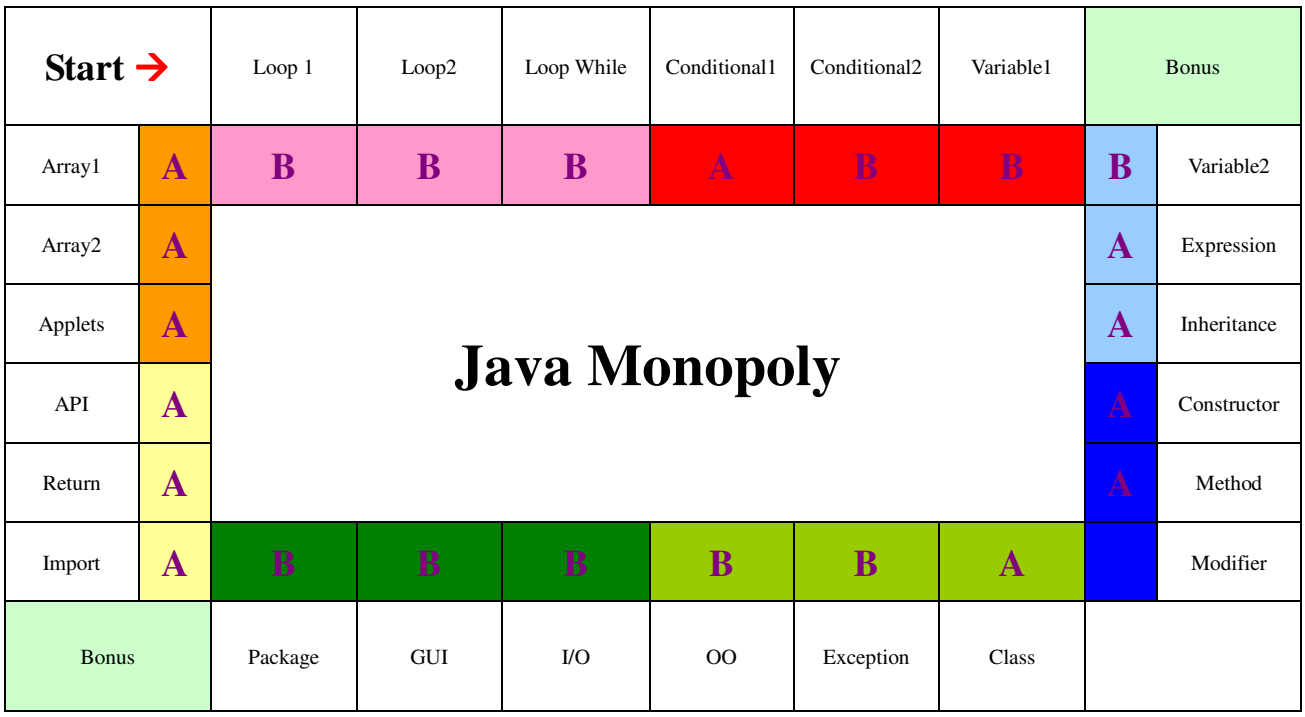

Figure 1: Cyber Java Monopoly Mockup; cells of A mean that Team A dominates

POSTER 23#

The Domain Concepts

$$P_A = A_1 + A_2 + A_3 \ldots + A_n \qquad P_B = B_1 + B_2 + B_3 \ldots + B_n \qquad \text{(Equation 1)}$$

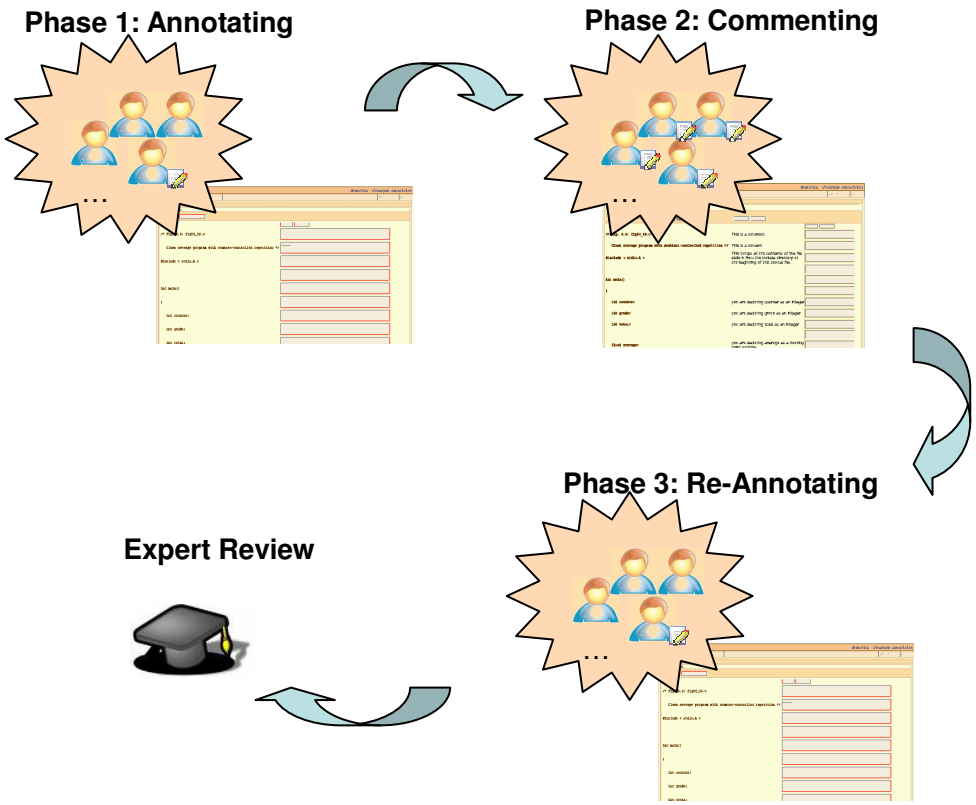

Figure 2: Student activities within the team

Phase 1 (Annotating): both teams' students have to give annotations for the examples separately in the current domain concept. Only the student with the Power of Editing has to do so.

Phase 2 (Commenting): students from each team can arbitrarily provide comments as feedback for their own teammates. Every student from the team is eligible to do so.

Phase 3 (Re-annotating): each team should submit the final version of example re-annotations based on the feedback from the second phase. Again, only the student with the Power of Editing has to do so.

Acquisition Score is the combination of the team's sum of Comment Score (C) and sum of Re-annotation Score(R) (Equation 2). Comments are derived from the contribution of the whole team. There are four types of comments; Praise, Agreeable,

Supplemental Annotation and Others (everything except the first three types is counted as type Others) (Equation 3). Re-annotation Score accounts for the correctness of the annotation lines. At present, the correctness of final annotations is graded by Experts.

$$AS = \sum_{i=1}^{n} C_i + \sum_{i=1}^{n} R_i \qquad \text{(Equation 2)}$$

$$C = 20\% \sum \Pr aise + 15\% \sum Agreeable + 55\% \sum Supplemental + 10\% \sum Others$$

ps. This is a sample composition (Equation 3)

IF $AS_A > AS_B$	IF $DC_A > DC_B$
DC = A	Team A wins
ELSE	ELSE
DC = B	Team A wins

Pseudo code of Assessment Logic: AS(Acquisition Score); DC(Domain Concept)

3. Preliminary Results

In the survey [9], the study participants explicitly say that they would like to be able to add their own annotations to the program code lines. Previously, we found that Re-annotation enhanced the annotation lines and annotating rate, it also improved the annotation quality [7][8]. To look further into the data, we found out the correlation between annotating rate and quality is high (r=0.83). It shows the more the annotations being re-annotated, the higher of the quality is. It encourages us to carry on using the mechanism to collaboratively author example annotations.

73% of our subjects agree and strongly agree that the annotating task helped them in their Problem Solving ability. None of them strongly disagree this argument. Due to the fact that Game Based Learning (GBL) is similar to Problem Based Learning (PBL) [5], this ratio basically supports our idea to apply game-based learning.

4. Summary & Future Work

Base on previous studies and our preliminary findings, we believe there's a strong demand for continuing using such mechanism to collaboratively author example annotations. In our attempt to motivate students to interact with peers and the system, we propose a novel method to combine game rules in existing structure. We would like to believe that game-based approach will arouse more promising interaction in collaborative programming language learning. The system proposed in this research, Cyber Java Monopoly, will not only stimulate interaction through the structural game phases, but also prevent teachers' overloading on game creation and construction. We contend that friendly game-based interface and easy rules can attract students to put more efforts on the programming learning, as well as allow teachers to pay attention to other pedagogical tasks. However, under our present design, the evaluation of final annotations is relied on experts' or teachers' determinations and recognitions. The assessment may still take experts and teachers time and efforts.

For future work, we plan to introduce a self-learning evaluation support for Cyber Java Monopoly to assist experts and teachers in grading the final annotations. The future solution will address on three dimensions: (1) analyzing students' annotations and extracting concepts of annotations, (2) mapping concepts between annotations and learning topic, (3) suggesting assessed score and training system on evaluation. We expect to create a self-learning solution which iteratively learns the evaluation knowledge between the system and experts' feedbacks could help teachers save a great deal of time and effort to identify students' learning condition and improve the assessment effectively.

5. Reference:

[1] Pivec M., (2007) Play and Learn: Potentials of Game-based Learning. Guest Editorial, *British Journal of Educational Technology*, Vol. 38, Issue 3, May 2007

[2] UniGame (2003) UniGame: Survey on online game-based learning, Deliverable 1.1, EU funded project, 2003, URL: http://www.unigame.net/html/case_studies/D1.pdf

[3] Carlile, S., Barnet, S., Sefton, A. & Uther, J. (1998) Medical problem based learning supported by intranet technology: a natural student centred approach. International Journal of Medical Informatics, 50, 1-3, 225-233.

[4] Morrison, J. (2004) Where now for problem based learning? The Lancet, 363, 9403, 174.

POSTER 23#

[5] Barrows, H. S. & Tamblyn, R. M. (1980) Problem-Based Learning: An Approach to Medical Education (Springer Series on Medical Education), New York, Springer.

[6] Pivec, M. & Dziabenko, O. (2004) Game-based learning framework for collaborative learning and student e-teamwork. e-mentor 2/2004; On-line ISSN 1731-7428 (English version), http://www.e-mentor.edu.pl/_xml/wydania/4/42.pdf

[7] Hsiao, I. & Brusilovsky, P. (2007) Collaborative Example Authoring System: The Value of Re-annotation based on Community Feedback, Proceedings of World Conference on E-Learning, E-Learn 2007,Quebec City, Canada, October 15-19, 2007, AACE

[8] Hsiao, I. (2008) (in press) The Role of Community Feedback on Example Annotating Process: an Evaluation on AnnotEx

[9] Yudelson, M. and Brusilovsky, P. (2005) NavEx: Providing Navigation Support for Adaptive Browsing of Annotated Code Examples. In: C.-K. Looi, G. McCalla, B. Bredeweg and J. Breuker (eds.) Proceedings of 12th International Conference on Artificial Intelligence in Education, AIED 2005, (Amsterdam, July 18-22, 2005). Amsterdam: IOS Press, pp. 710-717.